The multiple facets of the canonical direct implicational basis

Karell Bertet, Bernard Monjardet

Université Paris 1 – Panthéon - Sorbonne –

CERMSEM

UMR CNRS 8095

Cahiers de la MSE

The multiple facets of the canonical direct implicational basis

Karell BERTET, L3I

Bernard MONJARDET, CERMSEM

2005.52

Centre National de la Recherche Scientifique

Maison des Sciences Économiques, 106-112 boulevard de L'Hôpital, 75647 Paris Cedex 13

http://mse.univ-paris1.fr/Publicat.htm

ISSN : 1624-0340

The multiple facets of the canonical direct implicational basis

K. Bertet and B. Monjardet

L3I - Université de La Rochelle - av Michel Crépeau - 17042 La Rochelle
karell.bertet@iniv-lr.fr
CERMSEM - Maison des Sciences Economiques - Université Paris 1 - 106-112 bd de l'Hopital - 75013 Paris
bernard.monjardet@univ-paris1.fr

Abstract

Closure systems on a set S arises in many areas as relational databases, data-mining, formal concept analysis, articicial intelligence, logical programming or lattice theory. Implicational systems represents an efficient and convenient tool to deal with a closure system, and have been studied in various areas, with different terminology. This paper states the equality between five implicational systems issued from different works and satisfying various properties. The three main properties are the directness, canonical and minimality properties, thus the name *canonical direct implicational basis* given to this unique implicationnal system. This paper also gives the link between the canonical direct implicational basis and the Horn functions (via the prime implicants). It concludes by the necessity to compare more closely related works made independantly, and with different terminology in order to take advantage of the results really new.

Key words: implicational system, closure operator, closure system, canonical direct basis, lattice, Horn function.

1 Introduction

An *i*mplicational system defined on a finite set S is any binary relation Σ defined on the set $\mathcal{P}(S)$; when the subsets A and B of S are related by Σ, one writes $A \rightarrow_\Sigma B$. To any implicational system defined on S one can associate a closure operator φ defined on S, i.e an isotone, extensive and idempotent map from $\mathcal{P}(S)$ to $\mathcal{P}(S)$. In order to compute the closure $\varphi(X)$ of a subset X of S, one first computes the map $\pi_\Sigma(X) = X \cup \{B \subseteq S :$ there exists $A \subseteq X$ such that $A \rightarrow_\Sigma B\}$. The closure $\varphi(X)$ is obtained by iterating this map up obtaining a fixed point. The computation is clearly easier if Σ is *direct* in the sense that one gets always the closure of X without iteration, i.e. when for every $X \subseteq S$, $\varphi_\Sigma(X) = \pi_\Sigma(X)$. Conversely, given a closure operator φ (or the set of its closed sets), one can find implicational systems allowing to generate φ by the above construction.

Obviously, it is interesting to get either minimal such implicational systems or direct implicational systems. The above constructions and questions have appeared in several domains : theory of relational databases (minimal system of dependencies), data mining and formal concept analysis (descriptions and links between concepts), artificial intelligence and expert systems (extraction of rules), logical programming (Horn Boolean functions), lattice theory (join-irreducible representations in the lattice of closure operators) and have fostered many researches and results. This paper intends to show the equivalence of some of these results.

The second section recalls the notions used on posets, lattices, closure operators or closure systems, and (unary) implicational systems. In the third section we describe five implicational systems proposed by different authors in order to generate efficiently a closure operator (for reasons explained later they are called "bases" of the closure operator). The fourth section contains our main results. We prove that these five bases are the same and we define what can be called the canonical direct implicational basis. Whereas some of these equalities are easy to obtain, others are deduced from a non obvious characterization of a direct basis. One of the corollaries of these results shows the identity of the necessary sets for x (defined in the context of relational databases) and of the x-dominating sets (defined in the context of choice functions in microeconomics). It is (more or less) well known that closure systems on a set S are in a one-to-one correspondence with the so-called pure Horn Boolean functions defined on $\mathcal{P}(S)$. In the fifth section we show that finding the canonical direct implicational basis is the same that finding the prime implicants (or the prime implicates) of a (pure) Horn Boolean function. The sixth section is composed of two final notes: an historical note on the appearances of the notions considered in this paper; and an algorithmical note on the generation of a closure system, and the generation of the canonical direct implicationnal basis. The conclusion mentions some possible further researches, and, in particular, the need to compare more closely related works made independently in various domains in order to take advantage of the results really new.

2 Recalls and Definitions

2.1 Posets and Lattices

A *partially ordered set* $P = (S, \leq)$, also called a *poset*, is a set S[1] equipped with an order relation $\leq$ where an order relation is a binary relation which is reflexive ($\forall x \in S, x \leq x$), antisymmetric ($\forall x \neq y \in S$, $x \leq y$ imply $y \not\leq x$) and transitive ($\forall x, y, z \in S$, $x \leq y$ and $y \leq z$ imply $x \leq z$). We denote by $<$ the irreflexive relation associated to $\leq$, and by $\prec$ the *cover relation* defined by $x \prec y$ if $x < y$ and if there exists no $z \in S$ with $x < z < y$. We then say that x is *covered* by y (or y *covers* x). A poset P can also be given by its cover relation $\prec$ ($P = (S, \prec)$). The induced graphical representation is called the *(Hasse) diagram* of P. In the following, we will write indifferently $x \in S$ or $x \in P$. A poset $L = (S, \leq)$ is a *lattice* if any pair $\{x, y\}$ of elements of L has a *join* (i.e. a least upper bound) denoted by $x \vee y$ and a *meet* (i.e. a greatest lower bound) denoted by $x \wedge y$. Therefore, a lattice contains a minimum element (according to the relation $\leq$) called the *bottom* of the lattice, and denoted $\perp_L$ (or simply $\perp$). Respectively, a lattice contains a maximum element called the *top* of the lattice, and denoted $\top_L$ (or simply $\top$).

An element j (respectively, m) of a lattice L is a *join-irreducible* (respectively, *meet-irreducible*) of L if it cannot be obtained as the join (respectively, meet) of elements of L all distinct from j (respectively, from m). Equivalently, an element j (respectively, m) of L is join- (respectively, meet-) irreducible if it covers (respectively, is covered by) a unique element in L, which is then denoted by j^- (respectively, m^+) and called the *lower cover* of j (respectively, *upper cover* of m). The sets of join-irreducibles and of meet-irreducibles of a lattice $(L, \leq)$ are respectively denoted by J_L and M_L. For an element x in L, we denote by J_x (respectively, M_x) the set of all join-irreducibles j (respectively, meet-irreducibles m) such that $j \leq x$ (respectively, $x \leq m$).

Signifiant relations between the irreducible elements of a lattice L are given by the *arrow relations* defined on the set $J_L \times M_L$ as follows, with $j \in J_L$ and $m \in M_L$:

- $j \uparrow m$ if m is maximal in $\{t \in L : j \not\leq t\}$.
- $j \downarrow m$ if j is minimal in $\{t \in L : t \not\leq m\}$.

[1] In this paper, all the sets will be finite

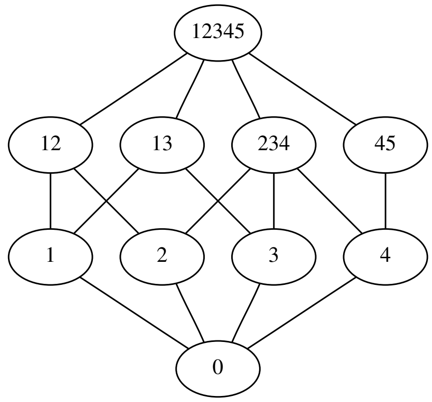

Fig. 1. The lattice $(\mathbb{F}, \subseteq)$ represented by its Hasse diagram, where $\mathbb{F}$ is the closure system of our example.

2.2 Set systems and Lattices

A *set system* on a set S is a family of subsets of S. A *closure system* $\mathbb{F}$ on a set S, also called a *Moore family*, is a set system stable by intersection and which contains S: $S \in \mathbb{F}$ and $F_1, F_2 \in \mathbb{F}$ implies $F_1 \cap F_2 \in \mathbb{F}$. The subsets belonging to a closure system $\mathbb{F}$ are called the *closed sets* of $\mathbb{F}$. The poset $(\mathbb{F}, \subseteq)$ is a lattice with, for each $F_1, F_2 \in \mathbb{F}$, $F_1 \wedge F_2 = F_1 \cap F_2$ and $F_1 \wedge F_2 = \bigcap\{F \in \mathbb{F} \,|\, F_1 \cup F_2 \subseteq F\}$. Moreover, any lattice L is isomorphic to the lattice of closed sets of a closure system ([6]). The simplest closure system representing L is defined on J_L: it is the set system $\{J_x \,|\, x \in L\}$.

Example 1 *Consider the closure system*[2]*, on the set* $S = \{1, 2, 3, 4, 5\}$*:*

$$\mathbb{F} = \{\emptyset, 1, 2, 3, 4, 12, 13, 45, 234, S\}$$

One can verify that it is stable by intersection. The lattice $(\mathbb{F}, \subseteq)$ *is represented by its Hasse diagram in Figure 1. We will use this example to illustrate several notions in this paper.*

A *closure operator* on a set S is a map φ on $\mathcal{P}(S)$ satisfying, $\forall X, Y \subseteq S$:

$$X \subseteq \varphi(Y) \Leftrightarrow \varphi(X) \subseteq \varphi(Y) \tag{1}$$

Equivalently, and more usually, a closure operator is defined as a map φ satisfying the three following properties: φ is *isotone* (i.e. $\forall X, X' \subseteq S$, $X \subseteq X' \Rightarrow \varphi(X) \subseteq \varphi(X')$), *extensive* (i.e. $\forall X \subseteq S$, $X \subseteq \varphi(X)$) and *idempo-*

[2] In this example as in the following, a subset $X = \{x_1, x_2, \ldots, x_n\}$ is written as the word $x_1 x_2 \ldots x_n$. Moreover, we abuse notation in the following and use $X + x$ (respectively, $X \setminus x$) for $X \cup \{x\}$ (respectively, $X \backslash \{x\}$), with $X \subseteq S$ and $x \in S$.

tent (i.e. $\forall X \subseteq S$, $\varphi^2(X) = \varphi(X)$). Still equivalently, a closure operator is an extensive map satisfying the *path-independence* property (i.e. $\forall X, Y \subseteq S$, $\varphi(X \cup Y) = \varphi(\varphi(X) \cup Y)$). $\varphi(X)$ is called the *closure* of X by φ. X is said to be *closed* by φ whenever it is a fixed point for φ, i.e. when $\varphi(X) = X$.

A subset B of S is a *basis* of F, with F closed set for φ, if $\varphi(B) = F$ and $\varphi(A) \subset \varphi(B)$ for every $A \subset B$ (in other words, B is a *minimal generating set* of F). A subset B of S is *free* if for every $x \in B$ $x \notin \varphi(B \backslash x)$. Or, equivalently, B is free if and only if $\varphi(A) \subset \varphi(B)$ for every $A \subset B$, or if and only if B is a basis of $\varphi(B)$. An element x of a subset X of S is an *extreme point* of X if $x \notin \varphi(X \setminus x)$. One denotes by $\mathbb{E}x_\varphi(X)$ or simply $\mathbb{E}x(X)$ the set of extreme points of X. Observe that X is free if and only if $\mathbb{E}x(X) = X$. A subset C of S is a *copoint* of $x \in S$ if C is a maximal subset of S such that $x \notin \varphi(C)$. It is well known that in the lattice $\mathbb{F}_\varphi$, the copoints of x are the closed sets C such that $\varphi(x) \uparrow C$, and that they are meet-irreducible.

Closure operators are in one-to-one correspondance with closure systems. On the first hand, the set of all closed elements of φ forms a closure system $\mathbb{F}_\varphi$:

$$\mathbb{F}_\varphi = \{F \subseteq S \,|\, F = \varphi(F)\} \tag{2}$$

Dually, given a closure system $\mathbb{F}$ on a set S, one defines the closure $\varphi_{\mathbb{F}}(X)$ of a subset X of S as the least element $F \in \mathbb{F}$ that contains X:

$$\varphi_{\mathbb{F}}(X) = \bigcap \{F \in \mathbb{F} \,|\, X \subseteq F\} \tag{3}$$

In particular $\varphi_{\mathbb{F}}(\emptyset) = \bot_{\mathbb{F}}$. Moreover for all $F_1, F_2 \in \mathbb{F}$, $F_1 \vee F_2 = \varphi_{\mathbb{F}}(F_1 \cup F_2)$ and $F_1 \wedge F_2 = \varphi_{\mathbb{F}}(F_1 \cap F_2) = F_1 \cap F_2$.

2.3 Unary Implicational System

An *Unary Implicational System* (UIS for short) Σ on S is a binary relation between $\mathcal{P}(S)$ and S: $\Sigma \subseteq \mathcal{P}(S) \times S$. An ordered pair $(A, b) \in \Sigma$ is called a Σ-*implication* whose *premise* is A and *conclusion* is b. It is written $A \rightarrow_\Sigma b$ or $A \rightarrow b$ (meaning "A implies b"). A subset $X \subseteq S$ *respects* a Σ-implication $A \rightarrow b$ when $A \subseteq X$ implies $b \in X$ (i.e. "if X contains A then X contains b").

$X \subseteq S$ is Σ-*closed* when X respects all Σ-implications, i.e $A \subseteq X$ implies $b \in X$ for every Σ-implication $A \rightarrow b$. The set of all Σ-closed sets forms a closure system $\mathbb{F}_\Sigma$ on S:

$$\mathbb{F}_\Sigma = \{X \subseteq S \,|\, X \text{ is } \Sigma\text{-closed}\} \tag{4}$$

Then, we can associate to Σ a closure operator $\varphi_\Sigma = \varphi_{\mathbb{F}_\Sigma}$ which defines the closure of a subset $X \subseteq S$ by [43] and [44]:

$$\varphi_\Sigma(X) = \pi_\Sigma(X) \cup \pi_\Sigma^2(X) \cup \pi_\Sigma^3(X) \cup \ldots \tag{5}$$

where

$$\pi_\Sigma(X) = X \cup \bigcup\{b \mid A \subseteq X \text{ and } A \rightarrow_\Sigma b\} \tag{6}$$

Remark that S being finite, the procedure in (5) terminates. Moreover, $\varphi_\Sigma(X) = \pi_\Sigma^n(X)$ with $n \leq |S|$ being the first integer such that $\pi_\Sigma^n(X) = \pi_\Sigma^{n+1}(X)$.

Now, consider a closure operator φ on S. Then the closed sets of φ coincide with the Σ-closed sets of the following UIS:

$$\Sigma_\varphi = \{X \rightarrow y \mid y \in \varphi(X) \text{ and } X \subseteq S\} \tag{7}$$

It is easy to see that Σ_φ satisfies the two following properties:

F1 $x \in X \subseteq S$ imply $X \rightarrow_{\Sigma_\varphi} x$.
F2 for every $y \in S$ and all $X, Y \subseteq S$, $[X \rightarrow_{\Sigma_\varphi} y$ and $\forall x \in X$, $Y \rightarrow_{\Sigma_\varphi} x]$ imply $Y \rightarrow_{\Sigma_\varphi} y$.

Unary IS statisfying properties F1 and F2 are called full UIS and are in one-to-one correspondance with closure operators, and thus with closure systems and lattices (via the set representation of lattices by the closure system $\{J_x \mid x \in L\}$).

When an IS Σ is not full, the full IS Σ_φ induced by the closure operator φ $(= \varphi_\Sigma)$ can be obtained by applying recursively rules F1 and F2 to Σ. Σ is then called a *generating system* for the full IS Σ_φ, and thus for the induced closure operator φ, the closure system $\mathbb{F}_\Sigma$, and the induced lattice $(\mathbb{F}_\Sigma, \subseteq)$. When some UISs Σ and Σ' on S are generating systems for the same closure system, they are called *equivalent* (i.e. $\mathbb{F}_\Sigma = \mathbb{F}_{\Sigma'}$).

As an illustration, let us introduce Σ_{free}, the generating system of Σ_φ composed of the subsets of S that also are free subsets:

$$\Sigma_{free} = \{X \rightarrow y \,:\, y \in \varphi(X) \backslash X \text{ and } X \text{ free subset of } S\} \tag{8}$$

An UIS Σ is called *direct* or *iteration-free* if for every $X \subseteq S$, $\varphi(X) = \pi_\Sigma(X)$ (see Equation (6)). An UIS Σ is *minimal* or *non-redundant* if $\Sigma \setminus \{X \rightarrow y\}$ is not equivalent to Σ, for all $X \rightarrow y$ in Σ. It is *minimum* if it is of least cardinality, i.e. if $|\Sigma| \leq |\Sigma'|$ for all UIS Σ' equivalent to Σ. A minimum UIS is trivially non-redondant, but the converse is false. Σ is *optimal* if $s(\Sigma) \leq s(\Sigma')$

for all UIS Σ' equivalent to Σ, where the *size* $s(\Sigma)$ of Σ is defined by:

$$s(\Sigma) = \sum_{A \to b \in \Sigma} (|A| + 1) \tag{9}$$

A minimal UIS is usually called a *basis* for the induced closure system (and thus for the induced lattice), and a *minimum basis* is then a basis of least cardinality.

An implication $X \to_\Sigma x$ with $x \in X$ is called *trivial*. An UIS is called *proper* if it doesn't contains trivial implications. When an UIS is not proper, an equivalent proper UIS can be obtained by applying the following rule:

F3 delete $A \to_\Sigma b$ from Σ when $b \in A$.

In this paper, all UISs will be considered to be proper UISs. Then, for instance, the full IS will means the proper full IS deduced from the full IS by applying F3.

In the litterature, an implicational system (IS for short) Σ can also be defined as a binary relation on $\mathcal{P}(S)$. A Σ-implication is then an ordered pair $(A, B) \in \Sigma$, written $A \to_\Sigma B$, with $A, B \in \mathcal{P}(S)$. An equivalent unary IS can be obtained by applying the following rule:

F4 replace $A \to_\Sigma B = \{b_1, b_2, \ldots, b_n\}$ by $A \to_\Sigma b_1$, $A \to_\Sigma b_2, \ldots$ and $A \to_\Sigma b_n$.

Dually, an equivalent IS can be obtained from an unary IS by applying recursively the following rule:

F5 replace $A \to_\Sigma B$ and $A \to_\Sigma B'$ by $A \to_\Sigma B \cup B'$.

Generating systems (also called *covers*) and bases can be also defined for US. In this case, there exists an unique minimum basis, called the *canonical basis* ([19]), enabling to get all the others minimum basis. We denote Σ_{can} the UIS deduced from this canonical basis by applying F4. Other definitions and bibliographical remarks can be found in the survey of Caspard and Monjardet in [9]. Just recall that in the domain of relationnal data bases, an implication is called a *functionnal dependency*.

Example 2 *Consider the closure system of our example given by the lattice $(\mathbb{F}, \subseteq)$ in Figure 1. The two equivalent UIS that are the UIS Σ_{can} deduced from the canonical basis and the generating system Σ_{free} are:*

$$\Sigma_{can} = \begin{cases} (1)\ 5 \to 4 \quad (2)\ 23 \to 4\ (3)\ 24 \to 3\ (4)\ 34 \to 2 \\ (5)\ 14 \to 2\ (6)\ 14 \to 3\ (7)\ 14 \to 5\ (8)\ 2345 \to 1 \end{cases}$$

$$\Sigma_{free} = \begin{cases} (\ 1\ )\ 5 \rightarrow 4 \quad (\ 2\ )\ 23 \rightarrow 4 \quad (\ 3\ )\ 24 \rightarrow 3\ (\ 4\ )\ 34 \rightarrow 2 \\ (\ 5\ )\ 14 \rightarrow 2\ (\ 6\ )\ 14 \rightarrow 3 \quad (\ 7\ )\ 14 \rightarrow 5\ (\ 8\ )\ 25 \rightarrow 1 \\ (\ 9\ )\ 35 \rightarrow 1\ (10)\ 15 \rightarrow 2 \quad (11)\ 35 \rightarrow 2\ (12)\ 15 \rightarrow 3 \\ (13)\ 25 \rightarrow 3\ (14)\ 123 \rightarrow 5\ (15)\ 15 \rightarrow 4\ (16)\ 25 \rightarrow 4 \\ (17)\ 35 \rightarrow 4\ (18)\ 123 \rightarrow 4 \end{cases}$$

Remark that Σ_{can} and Σ_{free} are proper UIS since for every implication conclusion is not included in premisse. Remark also that $\Sigma_{can} \not\subseteq \Sigma_{free}$ since the Σ_{can}-implication (8) not belongs to Σ_{free}. Concerning the direct property, it is clear that Σ_{free} is a direct UIS. To show, on the contrary, that Σ_{can} is not direct, consider the φ_Σ-closure of 15: $\pi_\Sigma(15) = 15 + 4$ by applying Σ_{can}-implication (1) and $\pi^2_\Sigma(15) = (15+4)+2+3$ by applying Σ_{can}-implications (5) and (6). Therefore $\varphi_\Sigma(15) \neq \pi_\Sigma(15)$.

3 Some interesting bases

In this section we are going to define several proper UIS which are generating systems for a given closure operator φ (equivalently for a given closure system $\mathbb{F}$) which can be the closure operator associated to a given UIS Σ. In the litterature on IS, the term basis is often used not only for minimal IS but also for IS satisfying various minimality criteria. We will do the same by defining five such bases.

3.1 The direct-optimal basis Σ_{do}

A number of problems related to closure systems, (thus closure operators or lattices) can be answered by computing closures of the kind $\varphi_\Sigma(X)$, for some $X \subseteq S$. According to the definition (see Eq.(5)) $\varphi(X)$ can be obtained given an UIS Σ by iteratively scanning Σ-implications: $\varphi(X)$ is initialized with X then increased with b for each implication $A \rightarrow_\Sigma b$ such that $\varphi(X)$ contains A. The computation cost depends on the number of iterations, and in any case is bounded by $|S|$. It is worth noticing that for direct (or iteration-free) UISs the computation of $\varphi(X)$ requires only one iteration, since $\varphi_\Sigma(X) = \pi_\Sigma(X)$. The *direct-optimal* property combines the directness and optimality properties:

Definition 3 *An IS Σ is* direct-optimal *if it is direct, and if $s(\Sigma) \leq s(\Sigma')$ for any direct IS Σ' equivalent to Σ.*

In [4], Bertet and Nebut show that a direct-optimal UIS is unique and can be obtained from any equivalent and proper UIS:

Proposition 4 *[4] The* direct-optimal basis Σ_{do} *is obtained from any equivalent and proper UIS Σ as follows:*

(1) first apply recursively the following rule[3] *to obtain a direct equivalent UIS:*
__F7__ for all $A \rightarrow_\Sigma b$ and $C + b \rightarrow_\Sigma d$ with $d \neq b$, add $A \cup C \rightarrow d$ to Σ
(2) then apply the **F3** *rule to obtain a proper UIS, and the following rule to minimize premisses of the Σ-implications:*
__F8__ for all $A \rightarrow_\Sigma b$ and $C \rightarrow_\Sigma b$, if $C \subset A$ then delete $A \rightarrow_\Sigma b$ from Σ.

Example 5 *Consider our example given by $(\mathbb{F}, \subseteq)$ in Figure 1. The basis Σ_{do} is:*

$$\Sigma_{do} = \begin{cases} (1)\,5 \rightarrow 4 & (2)\,23 \rightarrow 4 & (3)\,24 \rightarrow 3 & (4)\,34 \rightarrow 2 \\ (5)\,14 \rightarrow 2 & (6)\,14 \rightarrow 3 & (7)\,14 \rightarrow 5 & (8)\,25 \rightarrow 1 \\ (9)\,35 \rightarrow 1 & (10)\,15 \rightarrow 2 & (11)\,35 \rightarrow 2 & (12)\,15 \rightarrow 3 \\ (13)\,25 \rightarrow 3 & (14)\,123 \rightarrow 5 & & \end{cases}$$

Ona can verify that Σ_{do} is direct like Σ_{free}. Moreover, $s(\Sigma_{do}) < s(\Sigma_{free})$ and $\Sigma_{do} \subset \Sigma_{free}$.

3.2 The dependence relation's basis Σ_{dep}

The dependence relation's basis Σ_δ on S is issued from the *dependence relation* δ defined for a lattice, and introduced in [31] (see also [32]):

Definition 6 *The* dependence relation's basis Σ_δ *is:*

$$\Sigma_\delta = \{X + y \rightarrow x \,:\, x\delta_X y \text{ and } X \text{ is minimal for this property}\} \tag{10}$$

where the dependence relation δ_X *is defined on S, with $x, y \in S$ and $X \subset S$, by:*

$$x\delta_X y \text{ if and only if } x \notin \varphi(X),\ y \notin \varphi(X) \text{ and } x \in \varphi(X + y) \tag{11}$$

The dual relation of the relation δ_X has been considered in [3] where it is called *domination*. One can observe that the *dependence relation* δ on the lattice $(\mathbb{F}, \subseteq)$ is then given by $x\delta y$ if there exists $X \subseteq S \backslash \{x, y\}$ such that $x\delta_X y$ (so $\delta = \cup\{\delta_X,\ X \subset S\}$).

[3] when Σ is not proper, this rule has to be apply only when $b \notin A$ and $d \notin A \cup C$

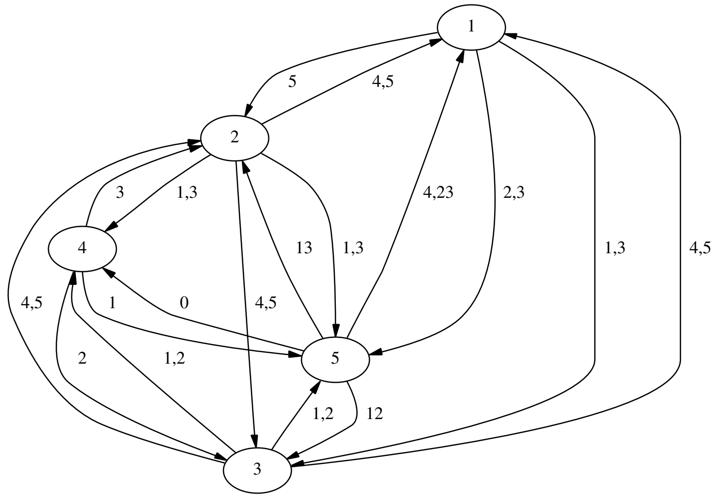

Fig. 2. Relation δ_X for $\mathbb{F}$ of our example represented by a directed graph where each relation $a\delta_X b$ is represented by an arc and labelled by X ($\emptyset$ is denoted by 0)

Example 7 *Figure 2 gives the dependence relations δ and δ_X of our example, where two vertices x and y are linked by an arc if $x\delta y$. This arc is valued by the subset X such that $x\delta_X y$. For instance, $5\delta_4 1$, and $5\delta_{23} 1$.*

3.3 The canonical iteration-free basis Σ_{cif}

The canonical iteration-free basis Σ_{cif} on S has been introduced by Wild in [44]. As for the direct-optimal basis it is based on the direct (also called iteration-free) property, that aims at efficiently generating closures $\varphi(X)$:

Definition 8 *[44] The* canonical iteration-free basis Σ_{cif} *is:*

$$\Sigma_{cif} = \{B \to x \,:\, x \in \varphi(B) \backslash \pi_\varphi(B) \text{ and } B \text{ is a free subset}\} \tag{12}$$

where π_φ is derived from φ as follows:

$\pi_\varphi(B) = B \cup \{x \in S :$ *there exists* $A \subset B$ *with* $x \in \varphi(A)$ *and* $\varphi(A) \subset \varphi(B)\}$ [4]

3.4 The left-minimal basis Σ_{lm}

The *left-minimal basis* Σ_{lm} is the restriction of the (proper) full UIS Σ_φ to implications where the premisse is of minimal cardinality. Using the definition of Σ_φ, Σ_{lm} can be expressed directly from φ:

[4] Note that the condition $\varphi(A) \subset \varphi(B)$ is useless when B is a free subset

Definition 9 *The* left-minimal basis Σ_{lm} *is:*

$$\Sigma_{lm} = \{X \rightarrow y \,:\, y \in \varphi(X) \setminus X \textit{ and for every } X' \subset X,\, y \notin \varphi(X')\} \quad (13)$$

An implication $X \rightarrow y$ is called *left-minimal* when it is a Σ_{lm}-implication. It is also called *proper implication* in [41] where implications are used in the data-mining area research, and *minimal functional dependency* in the domains of relational databases and Horn theories ([27,25]).

Example 10 *For our example, Σ_{lm} is the same has Σ_{do}. Remark that Σ_{lm} of our example has 14 implications, and not 15 as written wrongly in [9] on the same example (p.37).*

3.5 The weak-implication basis Σ_{weak}

The *weak-implication basis* has been introduced by Wille in [39] to show a connection between the theory of knowledge spaces ([13]) and formal concept analysis ([18]). It is based on the definition of a copoint (recall that a subset C of S is a copoint of $x \in S$ if C is a maximal subset of S such that $x \notin \varphi(C)$), and on the following classical notion of transversal set.

A subset B of a set S is a *transversal* of a family $\mathcal{F}$ of subsets of S if $B \cap F \neq \emptyset$ for every $F \in \mathcal{F}$. A transversal B is a *minimal transversal* of $\mathcal{F}$ if for every $A \subset B$, A is not a transversal of $\mathcal{F}$ (i.e. there exists $F \in \mathcal{F}$ with $A \cap F = \emptyset$).

Definition 11 *[39] The* weak-implication basis Σ_{weak} *is:*

$$\Sigma_{weak} = \{B \rightarrow x \,:\, B \subseteq S \textit{ and } B \textit{ is a blockade for } x\} \quad (14)$$

where a blockade *for $x \in S$ (also called x-*block*) is a minimal transversal of $\mathcal{D}_x$, the following family of subsets of S:*

$$\mathcal{D}_x = \{S \backslash (C + x)\,,\, C \textit{ is a copoint of } x\} \quad (15)$$

Lemma 12 *Let $x \in S$ and $B \subseteq S$. Then B x-block implies $x \notin B$ and $x \in \varphi(B)$ (i.e. $B \rightarrow x$)*

Proof Consider a x-block $B \subseteq S$. The first point is immediate: by definition of a blockade for x, we have $x \notin B$. For the second point, suppose $x \notin \varphi(B)$. Let $F \subseteq S$ closed set of φ such that $x \notin F$ and $\varphi(B) \subseteq F$. Then we have $\varphi(x) \uparrow F$. But $B \subseteq F$ implies $B \cap (S \backslash (F + x)) = \emptyset$, a contradiction with B a x-block. $\square$

4 The main results

The first result (Theorem 15) of this paper is to state the equality between the five bases defined in the previous section which thus are all direct bases. The second main result (Theorem 14) is to give an interesting characterization of the direct property based on an *exchange property.*

This exchange property has been independently introduced in [12] and under a stronger form in [4]. In [12], Demetrovics and Nam Son use it to define the notion of Sperner village and to show its equivalence with the notion of closure operator. In [4], Bertet and Nebut use it in the generation of the direct-optimal basis Σ_{do} where rule F7 is directly issued from this exchange property.

The characterisation of the Theorem 14 uses another formulation of the direct property issued from the definition (i.e. for every $X \subseteq S$, $\varphi(X) = \pi_\Sigma(X)$).

Lemma 13 *An UIS Σ is direct if and only if for every $X \subseteq S$, $\pi_\Sigma(X) = \pi^2_\Sigma(X)$.*

Theorem 14 *A proper UIS Σ is direct if and only if it satisfies the following* exchange condition*:*

$$\forall A, C \subseteq S\,,\, \forall b \in S \backslash A\,,\, \forall d \in S \backslash (A \cup C) \text{ and different from } b: \tag{16}$$

$$A \rightarrow_\Sigma b \text{ and } C + b \rightarrow_\Sigma d \text{ implies there exists } G \subseteq A \cup C \text{ such that } G \rightarrow_\Sigma d$$

Proof $\Rightarrow$: Let Σ be a direct UIS. Assume that for $b \in S\backslash A$, $d \in S\backslash(A \cup C)$ and different from b, we have $A \rightarrow_\Sigma b$ and $C+b \rightarrow_\Sigma d$, what means $b \in \varphi_\Sigma(A)$ and $d \in \varphi_\Sigma(C+b)$. Then, using the path-independence property of a closure operator, we get

$$d \in \varphi_\Sigma(A \cup (C+b)) = \varphi_\Sigma(\varphi_\Sigma(A+b) \cup C) = \varphi_\Sigma(\varphi_\Sigma(A) \cup C) = \varphi_\Sigma(A \cup C)$$

Now, Σ being direct, there exists $G \subseteq A \cup C$ such that $G \rightarrow_\Sigma d$.

$\Leftarrow$: Let Σ be an UIS satisfying condition (16). One must show that $\varphi_\Sigma(X) = \pi_\Sigma(X)$, or equivalently by Lemma 13 that $\pi_\Sigma(X) = \pi^2_\Sigma(X)$, or still equivalently (since π_Σ is extensive) that $\pi^2_\Sigma(X) \subseteq \pi_\Sigma(X)$.
Assume that there exists X with $\pi_\Sigma(X) \subseteq \pi^2_\Sigma(X)$, i.e. that there exists $z \in \pi^2_\Sigma(X)\backslash\pi_\Sigma(X)$. Then there exists $Z \subseteq \pi_\Sigma(X)$ with $Z \rightarrow_\Sigma z$ and $z \notin \pi_\Sigma(X)$. We set $p(Z) = |Z \cap (\pi_\Sigma(X)\backslash X)|$. The proof of $\varphi_\Sigma(X) = \pi_\Sigma(X)$ will follow immediatly from the proof of the following result:

if $p(Z) = p$ then there exists $Z' \subseteq S$ with $Z' \rightarrow_\Sigma z$ and $p(Z') < p(Z)$.

Indeed, by iteration of this result we would get some $Z^{(k)}$ with $Z^{(k)} \to_\Sigma z$ and $p(Z^{(k)}) = 0$, what means $Z^{(k)} \subseteq X$ and $z \in \pi_\Sigma(X)$, a contradiction with our hypothesis.
First, observe that $p(Z) > 0$: if not, $Z \in X$ and $z \in \pi_\Sigma(X)$, a contradiction. $p(Z) > 0$ means that there exists $y \in Z$ with $y \in \pi_\Sigma(X)\backslash X$. Thus there exists $Y \subseteq X$ with $Y \to_\Sigma y$. Now writing $Z = U + y$, we have $Y \to_\Sigma y$, $U + y \to_\Sigma z$ with $y \notin Y$ and (since $z \notin \pi_\Sigma(X)$) $z \notin Y \cup U$ and z different from y. So, by applying the exchange condition, we get that there exists $Z' \subseteq Y \cup U$ with $Z' \to_\Sigma z$. Moreover, since $p(Y \cup U) = p(Z) - 1$, we have $p(Z') < p(Z)$ like wanted. □

Now, let us give our other main result.

Theorem 15 *Let φ be a closure operator defined on a set S, and the five associated UISs above defined. Then*

$$\Sigma_{do} = \Sigma_{cif} = \Sigma_{dep} = \Sigma_{lm} = \Sigma_{weak}$$

Proof We prove first $\Sigma_{cif} = \Sigma_{dep} = \Sigma_{lm} = \Sigma_{weak}$ by proving $\Sigma_{cif} \subseteq \Sigma_{dep} \subseteq \Sigma_{lm} \subseteq \Sigma_{weak} \subseteq \Sigma_{cif}$. Then we prove $\Sigma_{do} = \Sigma_{lm}$

$\Sigma_{cif} \subseteq \Sigma_{dep}$: Let $B \to x$ be a Σ_{cif}-implication. That means that $x \in \varphi(B)\backslash\pi_\varphi(B)$ where B is free, i.e. $x \in \varphi(B)$ and $x \notin \varphi(A)$ for every $A \subset B$. Take any y in B. Since $B\backslash y \subset B$ and B is free, one has $x \notin \varphi(B\backslash y)$, $y \notin \varphi(B\backslash y)$ and (obviously) $x \in \varphi((B\backslash y)+y)$. If $X \subset B\backslash y$, $X+y \subset B$, and so $x \notin \varphi(X+y)$. Then $B\backslash y$ is minimal such that $x, y \notin \varphi(X)$ and $x \in \varphi(X + y)$, i.e. $B \to x$ is a Σ_{dep}-implication.

$\Sigma_{dep} \subseteq \Sigma_{lm}$: Let $B = X + y \to x$ be a Σ_{dep}-implication. Then $x \notin \varphi(X)$ and for every $Y \subset X$, $x \notin \varphi(Y + y)$. So $B \to x$ is a Σ_{lm}-implication.

$\Sigma_{lm} \subseteq \Sigma_{weak}$: Let $B \to x$ be a Σ_{lm}-implication. Let us first prove that B is a transversal of $\mathcal{D}_x = \{S\backslash(C + x),\ C \text{ copoint of } x\}$ before to prove that it is a minimal transversal. Since $x \notin B$, B is a transversal of $\mathcal{D}_x$ if and only if B is a transversal of $\mathcal{D}'_x = \{S\backslash C,\ C \text{ copoint of } x\}$. Suppose there exists C copoint of x such that $B \cap (S\backslash C) = \emptyset$ and so $B \subseteq C$. Then $\varphi(B) \subseteq C$ which implies $x \in C$, a contradiction with C copoint of x.
Suppose now that B is not a minimal transversal of $\mathcal{D}_x$., i.e. that it exists $Y \subset B$ with Y transversal of $\mathcal{D}_x$. Since B is left-minimal for the implication $B \to x$, we have $x \notin \varphi(Y)$. Then there exists a copoint C of x such that $Y \subseteq \varphi(Y) \subseteq C$. Therefore $Y \cap (S\backslash C) = \emptyset$, a contradiction with Y transversal of $\mathcal{D}_x$.

$\Sigma_{weak} \subseteq \Sigma_{cif}$: Let $B \to x$ be a Σ_{weak}-implication. That means that $x \in \varphi(B)\backslash B$ and B is minimal transversal of $\mathcal{D}_x = \{S\backslash(C + x),\ C \text{ copoint of } x\}$. We prove first that B is free by showing that for any $A \subset B$ one has $\varphi(A) \subset \varphi(B)$. Indeed, when $A \subset B$, A is not a transverasl of $\mathcal{D}_x$ and there exists a copoint C of x such that $A \cap (S \backslash C + x) = \emptyset$. So $A \subseteq C$ (since $x \notin A$) and $\varphi(A) \subseteq C$. But $x \notin C$ implies $x \notin \varphi(A)$ and so $\varphi(A) \subset \varphi(B)$. Moreover, we have just proved that $x \notin \varphi(A)$ for every $A \subset B$, i.e. that $x \notin \pi_\varphi(B)$. Finally, $B \to x$ is a Σ_{cif}-implication.

$\Sigma_{lm} = \Sigma_{do}$**:** To prove the equality $\Sigma_{lm} = \Sigma_{do}$, let us prove that Σ_{lm} is direct-optimal (since there is a unique direct-optimal basis). First we prove that Σ_{lm} is direct, i.e. that for every $A \subseteq S$, $\varphi(A) = A \cup \{x \in S :$ there exists $B \subseteq A$ with $B \to_{\Sigma_{lm}} x\}$. This is obvious since one can take for B a basis of $\varphi(A)$ such that $B \subseteq A$.

Now, let us prove that Σ_{lm} is direct-optimal. Consider a direct and equivalent UIS Σ. It is sufficient to prove that, when $B \to x$ is a Σ_{lm}-implication, it is also a Σ-implication. Assume that it is not the case. Since $B \to x$ is left-minimal, $A \to x \notin \Sigma$ for every $A \subset B$. Therefore, $x \notin \varphi(B) = B \cup \{x \in S :$ there exists $A \subseteq B$ with $A \to_\Sigma x\}$, a contradiction with Σ direct.

□

The above result justifies the following definition:

Definition 16 *The unique basis obtained in Theorem 15 is called the* canonical direct basis*, and is denoted* Σ_{cd}*.*

Theorem 14 induces others nice characterizations of the canonical direct basis:

Corollary 17 *Let φ be a closure operator. The canonical direct basis Σ_{cd} is the least basis of the set of all direct bases ordered by inclusion.*

Corollary 18 *An UIS Σ is the canonical direct basis if and only if it satisfies the two following properties:*

(1) for every $x \in S$, $B \to_\Sigma x$ and $B' \to_\Sigma x$, B and B' are incomparable.
(2) the exchange condition (16).

One can also observe that Corollary 17 is equivalent to the property of Σ_{cif} to be iteration-free in a canonical way, introduced in [44].

One can observe that the first property in Corollary 18 can equivalently be reformulated using the terminology of a Sperner family like in [12]: for every $x \in S$, the set $\mathcal{B}_x$ of all premisses of the Σ-implications $B \to_\Sigma x$ forms a Sperner family.

The fact that $\Sigma_{lm} = \Sigma_{weak}$ shows that the Sperner family $\mathcal{B}_x$ is the family of blockades of x, i.e. the family of minimal transversals of the family $\mathcal{D}_x = \{S \setminus (C + x) : C$ copoint of $x\}$. We show now that the *necessary sets for x*, and the *x-dominating sets* introduced in the litterature are the same that the sets $S \setminus (C+x)$. Mannila and Raiha ([28,29]) define a *necessary set* for x as a minimal transversal of $\mathcal{B}_x$. On the other hand, one finds in Aizerman and Aleskerov's book on choice functions ([1]) the definition of an *x-dominating set* as a subset T of S such that $x \in \mathbb{E}x_\varphi(S \setminus T)$ and $x \notin \mathbb{E}x_\varphi(U)$ for every U satisfying $S \setminus T \subset U$.

Corollary 19 *Let φ be a closure operator on S, $T \subseteq S$ and $x \in S \setminus T$. The three following conditions are equivalent:*

(1) T is a necessary set for x,
(2) there exists a copoint C of x such that $T = S \setminus (C+x)$,
(3) T is an x-dominating set.

Proof

$1 \Leftrightarrow 2$ Let us denote by $\mathcal{M}_x$ the family of necessary sets for x. By definition, $\mathcal{M}_x = Tr(\mathcal{B}_x)$, the family of minimal transversal of $\mathcal{B}_x$. And, as said above, $\mathcal{B}_x = Tr(\mathcal{D}_x)$ the family of minimal transversal of $\mathcal{D}_x = \{S \setminus (C+x) : C$ copoint of $x\}$. But, it is well known that, when $\mathcal{F}$ is a Sperner family, $Tr(Tr(\mathcal{F})) = \mathcal{F}$. Therefore $\mathcal{M}_x = Tr(\mathcal{B}_x) = Tr(Tr(\mathcal{D}_x)) = \mathcal{D}_x$.

$2 \Rightarrow 3$ If $T = S \setminus (C+x)$, one has $S \setminus T = C+x$. Since C is a maximal set such that $x \notin C$, $x \in \mathbb{E}x(S \setminus T)$, whereas if $U \supset S \setminus T = C+x$, then $U \setminus x \supset C$ and $x \notin \mathbb{E}x(U)$.

$3 \Rightarrow 2$ Let T be an x-dominating set. So, $x \in \mathbb{E}x_\varphi(S \setminus T)$, i.e. $\{x \in \varphi((S \setminus T) \setminus x))$. Now, if $U \in S \setminus T$, $U \setminus x \in (S \setminus T) \setminus x$ and $x \in \mathbb{E}x_\varphi(U)$ means that $x \in \varphi(U \setminus x))$. Thus $(S \setminus T) \setminus x) = (S \setminus T + x)$ is a maximal set such that $x \in \varphi(S \setminus T + x))$, i.e. a copoint C of x. Then $T = S \setminus (C+x)$, with C copoint of x.

$\square$

One can notice that the equivalence between 2 and 3 was proved in [33] but only for closure operators satisfying the antiexchange property.

5 The canonical direct basis and the Horn functions

It is well known that families of subsets of a set S are in a one-to-one correspondence with the Boolean functions defined on the Boolean algebra $\mathcal{P}(S)$. Indeed, one can associate to a family $\mathcal{F}$ of subsets of S its characteristic function $f_\mathcal{F}$:

$$f_\mathcal{F}(M) = \begin{cases} 1 & \text{if } M \in \mathcal{F} \text{ with } M \subseteq S \\ 0 & \text{if not} \end{cases} \tag{17}$$

And conversely, one can associate to a Boolean function f from $\mathcal{P}(S)$ to $\{0,1\}$ the following family of subets of S called the *models* or the *true points* of f:

$$\mathcal{F}_f = \{M \subseteq S \,:\, f(M) = 1\} \tag{18}$$

Observe that the set of all Boolean functions ordered by $f \leq g$ if $\mathcal{F}_f \subseteq \mathcal{F}_g$ is itself a Boolean algebra.

By considering dually the *false points*, one can provide another one-to-one correspondance between families on S and Boolean functions on $\mathcal{P}(S)$. In the following, we will prefer this second correspondance that associates to a Boolean fonction h the family $\mathcal{F}_h$ of its *false points* or its *counter-models*. Conversely, one can associate to a family $\mathcal{F}$ on S the Boolean function $h_{\mathcal{F}}$:

$$\mathcal{F}_h = \{M \subseteq S \,:\, h(M) = 0\} \tag{19}$$

$$h_{\mathcal{F}}(M) = \begin{cases} 0 & \text{if } M \in \mathcal{F} \text{ with } M \subseteq S \\ 1 & \text{if not} \end{cases} \tag{20}$$

A less known and still less used fact is that the closure systems on S are in a one-to-one correspondence with the Boolean functions called *pure (or definite) Horn functions* (see historical notes for references). Then, any result on closure systems (or closure operators or implicational systems) can be translated into results on Horn functions, and conversely. In this section we are going to do this translation for the canonical direct basis.

In order to define Horn functions we will recall some classical definitions and facts. We denote by $2 = (0, 1, \vee, \wedge, ')$ the Boolean algebra on two elements 0 and 1, with the two Boolean operations $\vee$ (called *disjonction or sum*) and $\wedge$ (called *conjunction or product*), and the unary operation $'$ (called *complementation*). A *Boolean function* of n (Boolean) variables is then a function on 2^n to 2. We denote by $S = \{x_1, x_2, ...x_n\}$ the set of these Boolean variables. We denote by $(x_1, x_2, ...x_n)$ a n-tuple of values 0 or 1 taken by these variables (then, take care that the same symbol x_i can represent the variable x_i or the value 0 or 1 taken by this variable according as it belongs either to a set or to a n-uple). The set 2^n of these n-uples is in one-to-one correspondance with the set 2^S of subsets of S (by the map $(x_1, x_2, \ldots, x_n) \rightarrow \{x_i \in S \,:\, x_i = 1\}$) where such a subset will also be called a *point*.

A variable x is called a *literal* whereas the complemented variable x' is called a *complemented literal*. A *conjunction* (respectively, a *disjonction*) of literals and complemented literals, where each variable, complemented or not, appears at most once is called a *term* (respectively, a *clause*). A conjunction like $x \wedge y' \wedge z$ will be generally written more simply $xy'z$.

Let f be a Boolean function on 2^n and $(x_1, x_2, ...x_n)$ a *true n-tuple* of f (respectively, a *false n-tuple* of f), i.e. a n-tuple such that $f(x_1, x_2, \ldots, x_n) = 1$ (respectively, $f(x_1, x_2, \ldots, x_n) = 0$). A *true point* (respectively *false point*) of f is a subset of S corresponding to a true n-tuple (respectively false n-tuple) of f. To a true (respectively false) n-tuple (or point) of f one associates the term $\wedge\{x_i : x_i = 1\} \wedge \{x'_i : x_i = 0\}$ (respectively, the clause $\vee\{x_i : x_i = 0\} \vee \{x'_i : x_i = 1\}$).

The sum (respectively, the product) of all these terms (respectively, clauses) constitutes the *canonical disjonctive normal form* (respectively, the *canonical conjonctive normal form*) denoted as the canonical DNF (respectively as the canonical CNF) of f. But, using the well known properties of a Boolean algebra (such that $x = x \vee x = xx = x \vee (xy) = x(y \vee x)$, $x \vee x' = 1$, $xx' = 0$), it is possible to get many others disjonctive or conjonctive normal forms representing f. Any two such normal forms representing the same Boolean function are called *equivalent.*

Example 20 *For instance, let us denote more simply the set of n Boolean variables by* $\{1, 2, ...n\}$ *and consider the Boolean function defined on* $\{1, 2, 3, 4, 5\}$ *by its canonical DNF:*

$$\begin{aligned} h = {} & 1'2'3'4'5' \vee 12'3'4'5' \vee 1'23'4'5' \vee 1'2'34'5' \vee 1'2'3'45' \vee \\ & 123'4'5' \vee 12'34'5' \vee 1'2'3'45 \vee 1'2345' \vee 12345 \end{aligned}$$

Then, since:

$$\begin{aligned} 1'2'3'4'5' \vee 12'3'4'5' \vee 1'23'4'5' \vee 123'4'5' &= 3'4'5' \\ \textit{and } 1'2'34'5' \vee 12'34'5' &= 2'34'5' \end{aligned}$$

an equivalent DNF for h is:

$$\begin{aligned} h = {} & (1'2'3'4'5' \vee 12'3'4'5' \vee 1'23'4'5' \vee 1'2'34'5' \vee 1'2'3'45') \\ & \vee (123'4'5' \vee 12'34'5') \\ & \vee 1'2'3'45 \vee 1'2345' \vee 12345 \\ = {} & 3'4'5' \vee 2'34'5' \vee 1'2'3'45 \vee 1'2345' \vee 12345. \end{aligned}$$

A classical problem (called the *Boolean function minimization problem*) is to find minimum DNF (or CNF) of a Boolean function, i.e. DNF (or CNF) using a minimum number of literals (other minimization problems using other criteria can be also considered). The first step for this research is to find the so-called prime implicants (respectively, prime implicates) of the Boolean function f. A *prime implicant* (respectively, a *prime implicate*) of f is a term t (respectively, a clause c) such that (in the order between Boolean functions),

$t \leq f$ (respectively, $f \leq c$) and is maximal (respectively, minimal) with this property. Indeed, a Boolean function f is always the sum of its prime implicants and the product of its prime implicates. But, it is generally possible to delete some implicants (respectively, implicates) in these expressions to get an equivalent more economical DNF or CNF. Then the second step consists in searching the expressions using the less of implicants (respectively, implicates). For an arbitrary Boolean function, the search of all its prime implicants (respectively, implicates) is a NP-complete problem.

We now define the so-called *pure (or definite) Horn functions.* Since we will consider only such Boolean functions, we will henceforth omit the word pure. A term is called *Horn* if it contains exactly one complemented literal. For instance, $34'5$ is a *Horn term.* A DNF is called Horn is all its terms are Horn. A Boolean function is called a *Horn function* if it can be represented by a Horn DNF. Now we have the following well known result (see Section 6.1):

Theorem 21 *A Boolean function h of n variables $x_1, x_2, \ldots x_n$ is a Horn function if and only if the set of its false points is a closure system on $S = \{x_1, x_2, \ldots x_n\}$.*

Remark. In the litterature one finds also another definition of a Horn function. A clause is called Horn if it contains exactly one literal. For instance, $1 \vee 2' \vee 4' \vee 5'$ is a *Horn clause.* A CNF is called Horn is all its clauses are Horn. A Boolean function is called a *Horn function* if it can be represented by a Horn CNF. This definition is not equivalent to the previous one. In fact, a Boolean function f is a Horn function in this second sense if and only if the complementary function f' (in the Boolean algebra of all Boolean functions) is Horn in the first sense. With this second definition, one has: *"a Boolean function is a Horn function if and only if the set of its true points is a closure system".*

Now we can give the relation between the prime implicants of a Horn function h and the canonical direct implicational basis Σ_{cd} of its associated closure operator. It is known that the prime implicants of a Horn function are Horn terms, and so we can write Bx' such a prime implicant, where B is the subset of S corresponding to the literals of this prime implicant. For completeness we give the proof of the following known result (see for instance [25]).

Proposition 22 *Let $S = \{x_1, x_2, \ldots, x_n\}$ be a set of elements, and:*

- *h be a Horn function of n variables on $\mathcal{P}(S)$;*
- *$\mathcal{F}_h$ the closure system defined on S by the false points of h;*
- *φ_h the associated closure operator on S;*
- *Σ_{cd} the corresponding canonical direct implicational basis.*

Then Bx' is a prime implicant of h if and only if $B \to x \in \Sigma_{cd}$.

Proof Let Bx' be a prime implicant of h and consider the implication $B \to x$. It belongs to Σ_φ since $h(\varphi_h(B)) = 0$ implies $Bx'(\varphi_h(B)) = 0$ and so $x \in \varphi_h(B)$. Let $A \subset B$. Since Ax' is not an implicant of h, there exists $X \subseteq S$ such that $Ax'(X) = 1$ and $h(X) = 0$. Then, $A \subseteq X \subseteq S \setminus x$ and $X \in \mathcal{F}_h$ what means that $x \notin \varphi_h(A)$. So, $A \to x \notin \Sigma_\varphi$ and $B \to x \in \Sigma_{cd}$.

Conversely, let $B \to x \in \Sigma_{cd}$ and consider the Boolean term Bx'. For $X \subseteq S$, we have $Bx'(X) = 1$ if and only if $B \subseteq X$ and $x \notin X$. Then $X \in \mathcal{F}_h$ and $h(X) = 1$, what shows $Bx' \leq h$. Moreover, $B \leq h$ since $B(\varphi_h(B)) = 1$ and $h(\varphi_h(B)) = 0$. Similary, if $A \subset B$, $Ax' \not\leq h$, since $Ax'(\varphi_h(A)) = 1$ and $h(\varphi_h(B)) = 0$. Then Bx' is a prime implicant of h. □

Corollary 23 *There is a one-to-one map between the set of prime implicants of a Horn function and the set of implications in the canonical direct basis of the closure operator corresponding to the Horn function.*

Remark. When one Consider the definition of a Horn function mentionned in the previous remark. One gets a one-to-one map between the set of prime implicates of the Horn function and the set of implications in the canonical direct basis of the corresponding closure operator.

Example 24 *For our example (Example 1), consider its closure system $\mathbb{F}$ defined on $S = \{1, 2, 3, 4, 5\}$, and its canonical direct UIS Σ_{cd} (equal to Σ_{do} given in Example 5). By Proposition 22, $\mathbb{F}$ is the closure system given by the false points of the following Horn function whose prime implicants are deduced from Σ_{cd}:*

$$\begin{aligned} h = {} & 54' \vee 234' \vee 243' \vee 342' \vee 142' \vee 143' \vee 145' \vee \\ & 251' \vee 351' \vee 152' \vee 352' \vee 153' \vee 253' \vee 1235' \end{aligned}$$

For instance, one can verify that $12 \in \mathbb{F}$ is equivalent to $h(12) = 0$; and $14 \notin \mathbb{F}$ is equivalent to $h(14) = 1$.

6 Notes

6.1 Historical note

Unary implicational systems have been considered at least as early as in a Hertz's paper ([22]) on logical deductive systems where they formalize "consequence" relations. The link with closure operators (or systems) appears for example in a Scott's paper ([40]) as well as in Doignon and Falmagne's work [13], authors which speak of *entailment* relations. This link appears also in Buchi ([8]) where its *dependence* relations are the dual form of UIS, i.e. relations contained in $S \times \mathcal{P}(S)$. On the other hand, Armstrong ([2]) have shown the one-to-one correspondance between closure systems and full implicational systems. This paper has fostered many developments in relationnal databases theory (see, for instance, [28,27]). Related works appeared in data analysis ([19,17,39]).

Birkhoff ([5]) do date back the origin of the notions of closure systems and closure operators to Moore's papers ([34,35] [5]). But it is probable that Moore's observations on the equivalence between these two notions would have been forgotten if these two concepts, under various names and in a more or less general way, have not played a significant role in the birth of the general topology as an axiomatic theory in the beginning of the last century. Many mathematicians (Alexander, Alexandroff, Frechet, Hausdorff, Kuratowski, Riesz, Sierpinski, Siskorski, Monteiro, Ribero, Appert, etc...) contributed to this creation, using systems of axioms based on several different primitive notions like derivation, neighbourood, surrounding, closed or open sets, closure or interior operators....On the other hand, in early works of Tarski (for instance [42], see also Martin and Pollard's book 1996) the consequence relation of a *logical deductive system* was defined as a closure operator on an infinite set S satisfying a *"finitary"* axiom. Other logicians (like Hertz ([22]), Scott ([40])) defined it as a binary relation between sets and, later, a one-to-one correspondence between Scott's *"informations systems"* and *"algebraic $\cap$-structures"* has been displayed (see Davey and Priestley ([10])). In the finite case, this correspondence becomes exactly the one-to-one correspondence between full implicational systems and closure systems. This last one has been first shown by Armstrong ([2]), who, in the context of relational data bases, called a full implicational system a *full family of functional dependencies.*

The name *Horn clause* comes from the logician Alfred Horn, who first pointed out the significance of such clauses in [24]. This attribution is sometimes contested. For instance Hodges ([23]) writes: *" Horn clause logic is a part of*

[5] In the first edition of Lattice Theory ([5]) Birkhoff thanks O.Öre to have communicated to him the references of works' Moore and he uses the term Moore family for a closure system. In his 1909 paper ([34]), Moore speaks in terms of property of a class of functions : *"Let a property satisfied by the class (of all functions) and by the greatest common subclass of subclasses satisfying it. Then this property is extensionnaly attainable in the sense that for every subclass S there exists a least extensive class containing S, given by the intersection of all subclasses containing S".*

first-order logic. It was first isolated by J.C.C. McKinsey in [30]. The name *Horn* is a historical accident. After McKinsey's paper, Alfred Tarski suggested investigating a more general class of sentences which are like Horn clauses except that they have arbitrarily many existential and universal quantifiers at the beginning. The sentences which Tarski described are now known as Horn sentences, because Tarski's colleague Alfred Horn ([24]) responded to Tarski's suggestion by showing that one of McKinsey's theorems is true for them too. This work of Horn is important in its own right, but it is not directly relevant to Horn clauses. (Henschen in [21] p.820 explains the name "Horn clause" by a result of Horn in [24] on Horn clauses; but the result is false, and it is not in [24]."

On the other hand, Dechter and Pearl in [11] write that : *"the equivalence between Horn functions and family of subsets closed by intersection appears to be a general folklore among many researchers, although we could not trace its precise origin."*

Apparently, the result attributed to Horn by Henshen and said *"false and not in Horn"* by Hodges is the following: *"... the class of Horn sets equals the class of sets of clauses which are true in a cross product of two interpretations if and only if they are true in each individual interpretation"*. But, in fact, Horn's 1943 paper deals with Horn terms (i.e. propositional terms containing at most one complemented literal) and its Lemma 7 amounts exactly to say that a Boolean function h is Horn (in the sense that it admits a Horn DNF) if and only if the family of its false points is closed by intersection. So, this result has a precise origin and, far to be an *"historical accident"*, the term Horn (Boolean) functions seems quite justified.

It is interesting to mention that the equivalence between Horn Boolean functions and families of sets closed by intersection was rediscovered independently by Flament in [14]. Flament, motivated by the need to generalize the well-known *Guttman's scale* in questionnaries'analysis, searched to find systems of implications between questions explaining the answers of subjects to queries. Then, he associates with these answers a Boolean algebra. And he writes (page 198): *"le protocole (i.e. the family of sets associated to the answers of subjects to a series of dichotomous questions) est fermé pour l'intersection si et seulement si aucune des PCU (= prime implicant) ne comporte plus d'une réponse négative"*.

6.2 Algorithmical note

Closure systems appears in many areas where we need efficient algorithms to handle them. So, in these areas, many (generally independant) works have been made to adress the many algorithmical problem raised. In particular, since these systems may have several "representations", a general problem is to provide algorithms to go from a representation taken as input to another taken as output. We will come back later on the notion of representation, but we consider first the two following important transformations problems:

- **Generation of the canonical direct basis** Σ_{cd} with an equivalent UIS Σ as input.
- **Generation of the closure operator** φ **or/and of the family** $\mathbb{F}$ **of closed sets** with either an UIS Σ or the basis Σ_{cd} as input.

Since $\mathbb{F}$ and Σ_{cd} are bounded by $2^{|S|}$ in the worst case, and by 1 in the best case, with a reasonable size in practice, these problems belong to the more general class of problems having an input of size n, and an output of size N bounded by 2^n. For this class of problems, a classical worst-case analysis makes them exponential, thus NP-complete in time and in space. Special analysis techniques, that are output-sensitive, have to be used to go beyond this result (see a survey in [36]).

Regarding the time-analysis, the idea is to extract the time complexity needed to generate only one element of the output (i.e. one implication or one closed set in our case). Two analysis are proposed to compute such a complexity, depending on the algorithm. The first analysis is called the *amortized complexity* and computes the "average cost" per element. It consists in extracting the *amortized cost* c per element from the general classical time complexity $O(cN)$ where N is the size of the output. When c can be bounded by a polynomial, we talk about a *polynomial amortized time algorithm.* The second analysis, called the *delay complexity*, considers the output as a sequence of generated elements. It consists then in more accurately computing the time between the generation of two element called the *delay cost.* A *polynomial delay algorithm* is then an algorithm with a delay cost bounded by a polynomial.

Regarding the space-analysis, one can introduce the *storage requirement* which is satisfied when the output has to be stored in memory. Hence, one can distinguish between *counting algorithms* (output is only counted, and not stored), *generation algorithms* (output is generated, sometimes it has to be stored) and *construction algorithms* (output is generated and stored).

Consider the special case where $\mathbb{F}$ contains all the subsets of S, and thus with a size equal to $2^{|S|}$ (($\mathbb{F}, \subseteq$) is then a boolean lattice). The generation of $\mathbb{F}$ consists then in generating all subsets of S, that can be performed in $O(1)$ per element using algorithms of constant amortized time or of constant delay. Moreover, since storage is not required by these algorithms, they have a non exponential space complexity.

6.2.0.1 Generation of Σ_{cd}.

Since Σ_{cd} has an exponential size in the worst-case, any generation algorithm has also to be analysed by considering the time-complexity per implication. Currently, there exists no algorithm with a polynomial generation per implication, using the storage requirement. Moreover, the existence of such a polynomial algorithm is still an open problem. Wild in [44] provides an algorithm with an IS Σ as input that has an exponential time complexity per implication. His algorithm computes an intermediate but larger UIS of exponential size in the worst case. Bertet and Nebut's algorithm in [4] also generates an intermediate and exponential but direct UIS Σ_d before minimizing it, and computes Σ_{cd} in $O(|S||\Sigma_d|^2)$ by applying rule F7 then rule F3 as described in Definition 4. Let us also mention in the area of data-mining the algorithm of Taouil and Bastide in [41] where the left-minimal implications are called *proper implications*. It has the same exponential time ans space complexity per implication.

6.2.0.2 Generation of φ and $\mathbb{F}$.

In [28], Mannila and Räihä propose the generation of a closure $\varphi(X)$ (algorithm *Linclosure*) in $O(|S|^2|\Sigma|)$, with a given Σ as input. This algorithm iteratively scans implications of an UIS Σ, and the computation cost depends on the number of iterations, in any case bounded by $|S|$. In order to practically limit this number while keeping the same complexity, Wild in [44] modifies this algorithm using additional and sophisticated data structures. It is worth noticing that for direct UIS, and thus with Σ_{cd}, the computation of $\varphi(X)$ requires only one iteration. Therefore, using Σ_{cd}, a closure $\varphi(X)$ is obtained by Bertet and Nebut in [4] in $O(|X||\Sigma_{cd}|)$ (when expressed with respect to X) or in $O(s(\Sigma_{cd}))$ (when expressed with respect to Σ_{cd}). They also propose in [4] an algorithm to generate the family $\mathbb{F}$ by computing some closures $\varphi(X)$. This algorithm has an exponential space-complexity since the closed sets have to be stored, and a time complexity in $O(|S|^2 + |S|c_\varphi)$ per element, where c_φ is the cost to generate one closure $\varphi(X)$. Therefore, their algorithm is in $O(|S|^3)$ using a given Σ as input, and an improvement in $O(|S|^2 + |S|s(\Sigma_{cd}))$ is obtained using the canonical direct basis Σ_{cd} as input. This improvement is due to the direct property of Σ_{cd} and can be applied to every algorithm that uses the computation of a closure $\varphi(X)$ as a basic step.

6.2.0.3 Others representations of $\mathbb{F}$.

When one takes the closure system $\mathbb{F}$ (or its associated closure operator φ) as the primitive structure, one can consider any implicationnal system Σ such that $\mathbb{F} = \mathbb{F}_\Sigma$ as a representation $\mathcal{R}_\mathbb{F}$ of it. Nevertheless, to take the full (unary) implicationnal system Σ_φ (given by Formula (7)) would be not very efficient. A representation $\mathcal{R}_\mathbb{F}$ of a closure system $\mathbb{F}$ is said good when it is small, readily identifiable, and when it uniquely determinates $\mathbb{F}$ via simple and efficient generation algorithm. This paper focuses on the representation of $\mathbb{F}$ by the canonical direct basis ($\mathcal{R}_\mathbb{F} = \Sigma_{cd}$).

However, one can find many others (but not always good) representations $\mathcal{R}_{\mathbb{F}}$ of a closure system $\mathbb{F}$: representation by the canonical basis in data analysis ([19]) ; representation by Horn functions in logical programming ([15]) ; representation by a poset of irreducibles in lattice theory ([36]) ; representation by a table called a *context* in formal concept analysis ([18]) and data-mining. Some links between these different representations have been studied. For instance, in [25], one finds the correspondance between left minimal implications (called there the minimal functionnal dependencies) and prime implicates (Corollary 23). When $\mathcal{R}_{\mathbb{F}}$ is one of these representations the two generation problems becomes the generation of the direct canonical basis Σ_{cd} with another representation $\mathcal{R}_{\mathbb{F}}$ as input, and the generation of a closure system $\mathbb{F}$ with $\mathcal{R}_{\mathbb{F}}$ as input.

Concerning the generation of Σ_{cd} from another representation, let us mention the algorithm of Mannila in [28], with the irreducibles elements of $\mathbb{F}$ as input. It has an exponential time per implication in the worst case, and is based on the generation of all minimal transversals, problem known to be exponential. Issued from the links between UIS and Horn functions, let us also mention the algorithm with the best known complexity that is due to Fredman and Khachiyan ([15]) with a DNF as input. It generates one implication in $O(|S|^{\log|S|})$, i.e a quasi-polynomial time and has recently been modified in [26] to solve this problem with a first step in deterministic polynomial time, following by $O(log^2|S|)$ non deterministic steps. Although it is computationnaly difficult to compute Σ_{cd} with a general DNF as input, one can compute a Σ_{cd}-implication in a polynomial time with a Horn DNF or the family $\mathbb{F}$ as input, using an algorithm due to Ibaraki et al. ([25]).

The well-known algorithm generating $\mathbb{F}$ is the *Next-closure* algorithm due to Ganter ([16]) in the context of the formal concept analysis ([18]). It accepts a table, and more generally a closure operator as input, and has a polynomial space-complexity (since the closed sets have not to be stored) and a time complexity in $O(|S|^3)$ per element. It generates closed sets according to a total order on all the closed sets (extending the inclusion order) called the *lectic order*. One can find various algorithms generating $\mathbb{F}$ using different representations, with the same complexity as the Next-closure algorithm, i.e. in $O(|S|^3)$ per generated closed set. However, the algorithm with the best known complexity uses a poset or a table as representation and is due to Nourine and Raynaud in [37]. It has a time-complexity in $O(|S|^2)$ per generated closed set, and an exponential space-complexity since all closed sets have to be stored in a tree structure (so it satisfies the storage requirement). Let us mention also an attribute-incremental algorithm generating the canonical basis from a context ([38]).

7 Conclusion

We have shown in this paper that several notions of bases for a closure operator (or an implicational system or a lattice of closed sets) are the same, and define what we have called the canonical direct implicational basis. One could also mention that finding this basis is equivalent finding in the lattice of all closure systems defined on S the expression of a closure system φ as the meet of the minimal meet-irreducible closure systems containing φ (see [9]). Moreover, the search of the canonical direct implicational basis is equivalent to the search of the prime implicants or the prime implicates of a Boolean function, a Horn function associated with the closure operator. So, the following problems, appeared in different domains, are all equivalent:

- *"Finding the canonical direct basis of a closure operator"*,
- *"Inferring minimal functional dependencies in a Horn theory"* or
- *"Computing the set of prime implicants of a Horn function"*.

Since equivalent notions as closure systems (or systems of sets closed by union), closure operators (or dual closure operators), full systems of implications (dependencies), Horn functions have been studied by different authors in different domains (topology, lattice theory, hypergraph theory, choice functions, relational data bases, data mining and concept analysis, artificial intelligence and expert systems, knowledge spaces, logic and logic programming, theorem proving...), it is not surprising that one finds the same notions, results or algorithms under various names. For instance, when Horn functions are used in artificial intelligence, the meet-irreducible elements of the associated lattice of closed sets are called its *characteristic sets*, and the the associated closure operator is called the *forward chaining procedure*. In the context of Horn functions, another interesting example is the significant directed graph introduced at least since 1987 by different authors on the set of the Boolean variables (see for instance [20,7]). It can be shown that in the case of a pure Horn function, the relation defined by this graph is the inverse of the dependency relation defined in Section 3 (it is also the *domination* relation in [3]).

On the other side, ono can find also many original results or algorithms but which are generally known only in a specified domain. It would be very profitable to extend (or create) the communications between the various domains using the same (or equivalent) tools. Our paper is a step in this direction.

References

[1] M.A. Aizerman and F.T. Aleskerov. *Theory of choice.* Elsevier Science B.V, North Holland, 1995.

[2] W.W. Armstrong. Dependency structures of database relationships. *Information Processing*, pages 580–583, 1974.

[3] A. Berry and E. San Juan. Generalized domination in closure systems. preprint, 2003.

[4] K. Bertet and M. Nebut. Efficient algorithms on the Moore family associated to an implicational system. *DMTCS*, 6(2), 2004.

[5] G. Birkhoff. *Lattice theory.* American Mathematical Society, 1st edition, 1940.

[6] G. Birkhoff and O. Frink. Representations of lattices by sets. *Transactions of the American Mathematical Society*, 64:299–316, 1948.

[7] E. Boros, O. Cepek, and A. Kogan. Horn minimization by iterative decomposition. *Annals of Mathematics and Artificial Intelligence*, 23(3-4):321–343, 1998.

[8] J.R. Buchi. *Finite Automatas, their Algebras and Grammars. Toward a theory of formal expressions.* Springer Verlag, Berlin, 1989.

[9] N. Caspard and B. Monjardet. The lattice of closure systems, closure operators and implicational systems on a finite set: a survey. *Discrete Applied Mathematics*, 127(2):241–269, 2003.

[10] B.A. Davey and H.A. Priestley. *Introduction to lattices and orders.* Cambridge University Press, 2nd edition, 1991.

[11] R. Dechter and J. Pearl. Structure identification in relational data. *Artificial Intelligence*, 58:237–270, 1992.

[12] J. Demetrovics and H. Nam-Son. Databases, closure operations and Sperner families. In P. Frankl et al., editor, *Extremal Problems for Finite Sets*, pages 199–203, New Haven, 1991. János Bolyai Mathematical Society.

[13] J.P. Doignon and J.C. Falmagne. *Knowledge spaces.* Springer Verlag, Berlin, 1999.

[14] C. Flament. *L'analyse booléenne des questionnaires.* Mouton, Paris, 1976.

[15] M.L. Fredman and L. Khachiyan. On the complexity of dualization of monotone disjunctive normal forms. *Journal of Algorithms*, 21:618–628, 1996.

[16] B. Ganter. Two basic algorithms in concept lattices. Technical report, Technische Hochschule Darmstadt, 1984.

[17] B. Ganter. Attribute exploration with background knowledge. *Theoretical Computer Science*, 217:215–233, 1999.

[18] B. Ganter and R. Wille. *Formal concept analysis, Mathematical foundations.* Springer Verlag, Berlin, 1999.

[19] J.L. Guigues and V. Duquenne. Familles minimales d'implications informatives résultant d'un tableau de données binaires. *Mathematiques & Sciences Humaines*, 95:5–18, 1986.

[20] P.L. Hammer and A. Kogan. Quasi-acyclic propositionnal Horn knowledgebases: optimal compression. *IEEE Transaction on Knowledge and Data Engineering*, 7(5):751–762, 1995.

[21] L.J. Henschen. Semantic resolution for Horn sets. *IEEE Transactions on Computers C-25*, pages 816–822, 1976.

[22] P. Hertz. Über axiomensysteme für beliebige satzsysteme. *Mathematische Annalen*, 101:457–514, 1927.

[23] W. Hodges. Logical features of Horn clauses. In M. Gabbay, C.J. Hogger, J.A. Robinson, and J. Siekmann, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming, Logical Foundations*, volume 1, pages 449–503, 1993.

[24] A. Horn. On sentences which are true of direct unions of algebras. *Journal of Symbolic Logic*, 16:14–21, 1951.

[25] T. Ibaraki, A. Kogan, and K. Makino. Functional dependencies in Horn theories. *Artificial Intelligence*, 108:1–30, 1999.

[26] D.J. Kavvadias and E.C. Stravropoulos. Monotone boolean dualization is in co-np[$log^2 n$]. *Information Processing Letter*, 85:1–6, 2003.

[27] D. Maier. *The Theory of Relational Databases.* Computer Sciences Press, 1983.

[28] H. Mannila and K.J. Räihä. *The design of relational databases.* Addison-Wesley, 1992.

[29] H. Mannila and K.J. Räihä. Algorithms for inferring functionnal dependencies from relations. *Data & Knowledge Engineering*, 12(1):83–99, 1994.

[30] J.C.C. McKinsey. The decision problem for some classes of sentences without quantifiers. *Journal of Symbolic Logic*, 8:61–76, 1943.

[31] B. Monjardet. Arrowian characterizations of latticial federation consensus functions. *Mathematical Social Sciences*, 20(1):51–71, 1990.

[32] B. Monjardet and N. Caspard. On a dependance relation in finite lattices. *Discrete Mathematics*, 165:497–505, 1997.

[33] B. Monjardet and V. Raderinirina. The duality between the anti-exchange closure operators and the path independent choice operators on a finite set. *Mathematical Social Sciences*, 41(2):131–150, 2001.

[34] E.H. Moore. On a form of general analysis with applications to linear differential and integral equations. In *Atti del IV Congr. Int. dei Mat. II*, pages 98–114, Roma, 1909.

[35] E.H. Moore. *Introduction to a form of general analysis.* Yale University Press, New Haven, 1910.

[36] L. Nourine. *Une structuration algorithmique de la théorie des treillis.* PhD thesis, Université of Montpellier I, July 2000.

[37] L. Nourine and O. Raynaud. A fast algorithm for building lattices. *Information Processing Letters*, 71:199–204, 1999.

[38] S. Obiedkov and V. Duquenne. Incremental construction of the canonical implication basis. In *Fourth International Conference Journées de l'Informatique Messine*, pages 15–23, 2003. submitted to Discrete Applied Mathematics.

[39] A. Rush and R. Wille. Knowledge spaces and formal concept analysis. In H.H. Bock and W. Polasek, editors, *Data Analysis and Information Systems*, pages 427–436, Berlin, 1995. Springer Verlag.

[40] D.S. Scott. Domains for denotational semantics. In *Automata, Languages and Programming*, volume 140, pages 577–613, Berlin, 1982.

[41] R. Taouil and Y. Bastide. Computing proper implications. In 9^{th} *International Conference on Conceptual Structures*, Stanford, USA, 2002.

[42] A. Tarski. Fundamentale begriffe der methodologie der deductiven wissenchaften. *Monatshelfte Mathematik Physik*, 37:361–404, 1930.

[43] M. Wild. A theory of finite closure spaces based on implications. *Advances in Mathematics*, 108(1):118–139, 1994.

[44] M. Wild. Computations with finite closure systems and implications. In *Proceedings of the 1st Annual International Conference on Computing and Combinatorics*, volume 959 of *LNCS*, pages 111–120. Springer, 1995.